ESANN 2021 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Online event, 6-8 October 2021, i6doc.com publ., ISBN 978287587082-7. Available from http://www.i6doc.com/en/.

Multivariate Time Series Multi-Coclustering. Application to Advanced Driving Assistance System Validation

Etienne Goffinet[1,2], Mustapha Lebbah[1], Hanane Azzag[1],
Loïc Giraldi[3] and Anthony Coutant[4]

1- Sorbonne Paris-Nord University - LIPN-UMR 7030, Villetaneuse - France
2- Groupe Renault SAS Avenue du Golf, Guyancourt - France
3- CEA, DES, IRESNE, DEC, 13108 Saint-Paul-Lez-Durance, France
4- HephIA, 30, rue de Gramont, Paris - France

Abstract. Driver assistance systems development remains a technical challenge for car manufacturers. Validating these systems requires to assess the assistance systems performances in a considerable number of driving contexts. *Groupe Renault* uses massive simulation for this task, which allows reproducing the complexity of physical driving conditions precisely and produces large volumes of multivariate time series. We present the operational constraints and scientific challenges related to these datasets and our proposal of an adapted model-based multiple coclustering approach, which creates several independent partitions by grouping redundant variables. This method natively performs model selection, missing values inference, noisy samples handling, confidence interval production, while keeping a sparse parameter numbers. The proposed model is evaluated on a synthetic dataset, and applied to a driver assistance system validation use-case.

1 Introduction

Advanced driver-assistance systems (ADAS) are automated embedded systems that bring support to driving tasks (e.g., emergency braking, adaptative cruise control, automated lighting, navigation, . . .). Given the high number of different vehicles, driving conditions, traffic laws, and given the expected reliability, it is today impossible to validate ADAS exhaustively only with physical tests "on tracks". *Groupe Renault* has made the technical choice to invest in massive driving simulation technology to circumvent this issue. For a given use case, a simulation protocol produces a multivariate time series dataset with varying dimensions: from 1E2 to 1E6 observations, and from 1E1 to 1E3 variables. The behavior of the simulated car cannot always be predicted. In consequences, it is impossible to use supervised methods. Our objective is to provide the ADAS developer with a synthetic view of the simulated dataset, simultaneously discriminating driving patterns and dependent sensors.

The model-based clustering methods [1] are a family of unsupervised probabilistic approaches that natively provide probabilistic estimations of cluster memberships, confidence intervals for probabilistic outlier detection, can infer missing values and model sample noise. In addition, they can handle data sets of varying sizes in opposition to existing Deep Learning clustering approaches [2, 3] that require massive amounts of data. An instance of such model-based approach has been used recently in the ADAS validation context [4], but restricted to univariate time series analysis. In a multivariate context, the possible relations between variables introduces additional difficulties. In most of real-life use cases, it is likely that each variable is associated to a different partition of the observation space (called *row-partition*). In the following, we assume the presence of a variable partition (called *redundant column-partition*), such that variables in a redundant column-cluster share the same row-partition. In our use cases, this assumption seems rational, as we know that many variables are related (e.g., position and acceleration, ADAS activation and braking). The *Multi-Clustering* [5] designates a set of methods that infers this multiple-partitions representation of a dataset, as illustrated in Fig. 1 - b). In a *parametric* model-based context [6, 7], using these approaches requires to know the partitions sizes a priori (rarely true in practice), or to perform a model selection step. However, the combinatorial nature of Multi-Clustering makes the model selection a complex task (there are, for example, 184755 different multi-clustering models with at most 10 clusters of variables and 10 clusters of observations). Heuristic approaches could be considered at the cost of assumptions on model structure (e.g. with greedy strategies). The *non-parametric* approach circumvents this issue by natively integrating a model selection. A non-parametric Bayesian Multi-Clustering model has been developed by [8] for continuous datasets. In this model, the rows belonging to a block follow an independent multivariate distribution (c.f. Fig. 1-b)). This Multi-Clustering method groups variables according to their row partition but cannot regroup variables with similar distributions. Moreover, estimating the block distributions parameters becomes problematic when the column cluster sizes are high (e.g., high dimensional covariance matrices). Each redundant col-

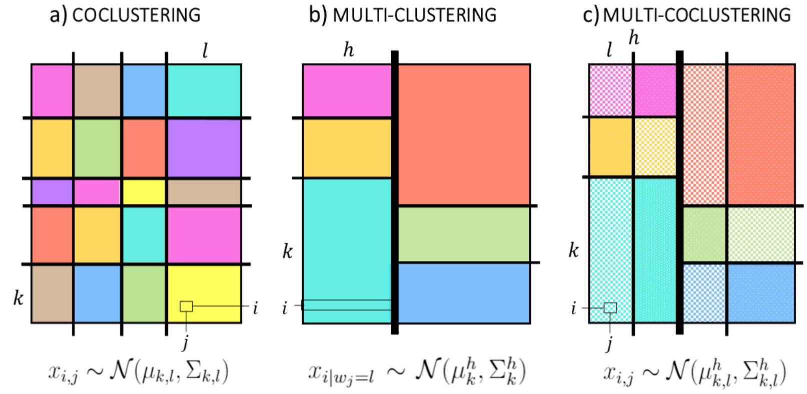
a) COCLUSTERING
b) MULTI-CLUSTERING
c) MULTI-COCLUSTERING
l
h
l
h
k
i
j
k
i
k
i
j
$x_{i,j} \sim \mathcal{N}(\mu_{k,l}, \Sigma_{k,l})$
$x_{i|w_j=l} \sim \mathcal{N}(\mu^h_k, \Sigma^h_k)$
$x_{i,j} \sim \mathcal{N}(\mu^h_{k,l}, \Sigma^h_{k,l})$

Fig. 1: Coclustering, Multi-Clustering and Multi-Coclustering, with k, l, h the row-cluster, correlated cluster and redundant cluster indices (respectively). Color and pattern designate block membership.

umn cluster is modeled with a coclustering structure to deal with this problem.

The model-based coclustering [9] (Figure 1 - a)) infers one row-partition and one column partition. This model assumes that all variables share the same row partition and that there exists groups of variables with common distribution. This assumption seems also natural, because our simulated variables are physically correlated (e.g., car speed and wheel rotation speed).

In this paper, we propose a new Non-Parametric Multi Coclustering (MCC) (c.f. Fig. 1-c) that combines the two previous approaches and produces the best of both worlds: the redundant partition layer extracts the true sparse number of row partition while the coclustering layers reduce the parameter set dimension and regroup correlated variables. To the best of our knowledge, there is only one comparable work by [10], which does not apply to multivariate time series.

2 Multi Coclustering of multivariate time series

2.1 MCC model definition

As a preprocessing step, the time series are transformed with a functional PCA [11] that first projects the time series in a common polynomial basis (in our case, an interpolated log-scaled Fourier periodogram [12]), then performs a PCA that produces a condensed representation. This preprocessing outputs a dataset $X = (x_{i,j,s})_{n \times p \times d}$, with n observations described by p variables, and d the projection space dimension (i.e., the number of PCA axes kept). We denote $\mathbf{v}$ the redundant partition indicator, $(\mathbf{w}_h)_h$ the correlated partition indicator, and Z the row-partitions indicator matrix. The model is formally defined by:

$$x_{i,j} \mid \{v_j = h, w_j^h = l, z_i^h = k, \theta_{k,l}^h\} \sim \mathcal{N}(\theta_{k,l}^h),$$

$$\theta_{k,l}^h \sim G_0,\ v_j \sim Mult(\eta),\ w_j^h \sim Mult(\rho_h),\ z_i^h \sim Mult(\pi_h),$$

$$\eta_j(\mathbf{r}) = r_j \prod_{j'=1}^{j-1} (1 - r_{j'}), \quad r_j \overset{\text{i.i.d.}}{\sim} Beta(1,\gamma)\ \ , \gamma \sim Gamma(a_r, b_r)$$

$$\rho_j^h(\mathbf{s}^h) = s_j^h \prod_{j'=1}^{j-1} \left(1 - s_{j'}^h\right), \quad s_j^h \overset{\text{i.i.d.}}{\sim} Beta(1,\beta_h)\ , \beta_h \sim Gamma(a_c, b_c)$$

$$\pi_j^h(\mathbf{t}^h) = t_j^h \prod_{j'=1}^{j-1} \left(1 - t_{j'}^h\right), \quad t_j^h \overset{\text{i.i.d.}}{\sim} Beta(1,\alpha_h),\ \alpha_h \sim Gamma(a_l, b_l),$$

where each memberships proportions vector η, $(\rho^h)_h$ and $(\pi^h)_h$ follows a *stick-breaking* construction scheme [13] with Gamma distribution hyperprior. The block distributions are multivariate normal and G_0 is the associated conjugate Normal-inverse-Wishart prior.

2.2 Inference

We propose an iterative two-steps inference: 1) update of $\mathbf{v}$ given Z; 2) update of Z and $\mathbf{w}$ given $\mathbf{v}$. In the first step, each membership v_j is sampled sequentially according to:

$$p(v_j \mid \mathbf{v}_{-j}, Z, \mathbf{x}_{.,j}, \chi) \propto \begin{cases} \dfrac{p_h}{p-1+\gamma}\,\mathrm{p}(\mathbf{x}_{.,j} \mid \mathbf{z}^h, \chi), & \text{existing cluster } h, \quad (1) \\ \dfrac{\gamma}{n-1+\gamma}\,\mathrm{p}(\mathbf{x}_{.,j} \mid \chi), & \text{new cluster,} \quad (2) \end{cases}$$

with $\mathbf{v}_{-j} = \{v_i : i \neq j\}$, $p(\mathbf{x}_{.,j} \mid \mathbf{z}^h)$ the distribution of $x_{.,j}$ in $\mathbf{z}^h$ and χ the hyperparameter set. In Eq. (1), the predictive distributions $\mathrm{p}(\mathbf{x}_{.,j} \mid \mathbf{z}^h, \chi)$ reduces to multivariate t-student densities product [14] thanks to the conjugacy of prior G_0. In Eq. (2), $\mathrm{p}(\mathbf{x}_{.,j} \mid \chi)$ is estimated with an univariate Dirichlet Process Mixture (once per variable, prior to the inference of the MCC), which also produces the optimal $\hat{\mathbf{z}}^j$ row partition associated with the new cluster. After $\mathbf{v}$'s update, the concentration parameter γ is updated based on [15]. In the second step, an NPLBM [16] is inferred on each sub-matrix $(X^h)_h$ defined by $\mathbf{v}$ with a Gibbs algorithm that simulates $\mathrm{p}(\mathbf{z}^h, \mathbf{w}^h \mid \mathbf{v}, X^h, \chi)$ (c.f. [16]).

3 Applications

3.1 Experiments on synthetic data

We compare the MCC's performances to other methods adapted to the use case constraints. The experiments dataset is composed of observations generated from a ground truth row-partition, that we seek to estimate. This ground truth dataset is "corrupted" by adding a proportion of other variables, uninformatives (generated from a one-cluster partition) or misleading (generated from another row-partition than the ground truth), noted p_u and p_m. The baselines methods are non-parametric model-based approaches to Multi-Clustering: Dirichlet Process Mixture model (DPM) that infers a row-partition with only one column-cluster; non-parametric coclustering method (CC) [16] displayed in Fig. 1; decoupled Multi-Clustering based on a two-layers nested DPM (NDPM); and non-parametric Multi-clustering (MCD) equivalent to MCC without the coclustering layer. The results, evaluated with the popular Rand Index (RI) and Adjusted Rand Index (ARI) (which are clustering quality metrics between -1 and 1, the higher the better), are displayed in Table 1 and show the interest of MCC in the presence of uninformatives or misleading variables.

3.2 Emergency lane Keeping assist use case

After validating MCC on a synthetic dataset, we apply it on a real ADAS validation dataset. In a straight lane scenario, the vehicle under test is drifting towards an oncoming car on the other lane. The ELK system is expected to detect the drifting, the oncoming car and avoid the collision with an emergency maneuver. The simulation generates $n = 500$ simulations described by $p = 150$ variables, totalling 75000 time series. The objective is to discriminate driving patterns, relevant groups of sensors and highlight the ADAS activation context. The Fig. 2-a) shows the MCC partitions overlaid on the dataset, with color

Table 1: Row-partition quality described by ARI, RI and row-cluster number $\hat{K}$

p_u	p_m	Scores	DPM	CC	NDPM	MCD	MCC
0	0	RI	0.84	**1.0**	0.57	**1.0**	**1.0**
0	100%		0.73	0.85	0.64	0.99	**1.0**
0	200%		0.79	0.85	0.76	0.88	**1.0**
100%	0		0.80	**1.0**	0.71	0.99	**1.0**
200%	0		0.51	**1.0**	0.60	0.98	**1.0**
0	0	ARI	0.67	**1.0**	0.27	**1.0**	**1.0**
0	100%		0.34	0.61	0.27	0.97	**1.0**
0	200%		0.47	0.61	0.39	0.70	**1.0**
100%	0		0.56	**1.0**	0.44	0.99	**1.0**
200%	0		0.13	**1.0**	0.27	0.96	**1.0**
0	0	$\hat{K}$	**3**	**3**	2	**3**	**3**
0	100%		7	6	**3**	**3**	**3**
0	200%		6	6	5	6	**3**
100%	0		**3**	**3**	2	**3**	**3**
200%	0		**3**	**3**	2	**3**	**3**

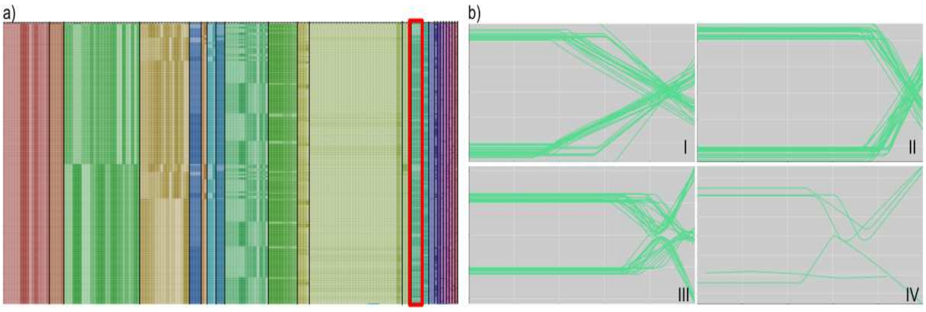

Fig. 2: a) MCC results. b) Car trajectories in the highlighted clusters

and transparency respectively indicating redundant column clusters and block partitions. Fig. 2-b) shows the car trajectories (outlined by a red rectangle in Fig. 2-a)) in the row-clusters: the sub-graph I shows a late ADAS activation scenario, in sub-graph II the ELK did not activate. In sub-graph III, the systems correctly change the car trajectory and prevent the collision. The last graph IV shows outlier simulations detected using the probability density of the MCC model. In addition to discriminating the non-informative variables (leftmost columns in red color), Fig. 2-A) shows a hierarchical structure between several row partitions (leftmost "green", "ochre" and "blue" columns). These partitions are linked to the vehicle orientation (e.g., road angle or steering wheel angle), lane position and ADAS activation indicators. Based on the results, the next steps for the ADAS developer consists in relating the partitions to the input simulation parameters and assessing the ADAS compliance with its specifications.

ESANN 2021 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence
and Machine Learning. Online event, 6-8 October 2021, i6doc.com publ., ISBN 978287587082-7.
Available from http://www.i6doc.com/en/.

4 Conclusion

This paper describes a new Bayesian non-parametric based method designed for the exploration of multivariate time series datasets produced by driving simulations. This solution infers a multi-level dependency structure that highlights the relationships between sensors and discriminates driving simulations patterns. This method can be used in other domains, as long as the corresponding structure dependency assumptions hold. In the future we consider combining the multivariate time-series multi-coclustering with the classification of the simulation parameters in order to identify ADAS failure and success contexts.

References

[1] Geoffrey J McLachlan, Sharon X Lee, and Suren I Rathnayake. Finite mixture models. *Annual review of statistics and its application*, 6:355–378, 2019.

[2] Naveen Sai Madiraju, Seid M Sadat, Dimitry Fisher, and Homa Karimabadi. Deep temporal clustering: Fully unsupervised learning of time-domain features. *arXiv preprint arXiv:1802.01059*, 2018.

[3] Florent Forest, Alex Mourer, Mustapha Lebbah, Hanane Azzag, and Jérôme Lacaille. An invariance-guided stability criterion for time series clustering validation. In *International Conference on Pattern Recognition (ICPR)*, 2020.

[4] Etienne Goffinet, Mustapha Lebbah, Hanane Azzag, and Loic Giraldi. Autonomous driving validation with model-based dictionary clustering. In *European Conference on Machine Learning, (ECML PKDD)*, pages 323–338. Springer International Publishing, 2021.

[5] Juhua Hu and Jian Pei. Subspace multi-clustering: a review. *Knowledge and information systems*, 56(2):257–284, 2018.

[6] Matthieu Marbac and Vincent Vandewalle. A tractable multi-partitions clustering. *Computational Statistics & Data Analysis*, 132:167–179, 2019.

[7] Vincent Vandewalle. Multi-partitions subspace clustering. *Mathematics*, 8(4):597, 2020.

[8] Yue Guan, Jennifer G Dy, Donglin Niu, and Zoubin Ghahramani. Variational inference for nonparametric multiple clustering. In *MultiClust Workshop, KDD-2010*, 2010.

[9] Gérard Govaert and Mohamed Nadif. *Co-clustering: models, algorithms and applications.* John Wiley & Sons, 2013.

[10] Tomoki Tokuda, Junichiro Yoshimoto, Yu Shimizu, Go Okada, Masahiro Takamura, Yasumasa Okamoto, Shigeto Yamawaki, and Kenji Doya. Multiple co-clustering based on nonparametric mixture models with heterogeneous marginal distributions. *PloS one*, 12(10):e0186566, 2017.

[11] JO Ramsay and BW Silverman. Principal components analysis for functional data. *Functional data analysis*, pages 147–172, 2005.

[12] Jorge Caiado, Nuno Crato, and Daniel Peña. Comparison of times series with unequal length in the frequency domain. *Communications in Statistics, Simulation and Computation*, 38(3):527–540, 2009.

[13] Jayaram Sethuraman. A constructive definition of dirichlet priors. *Statistica sinica*, pages 639–650, 1994.

[14] Andrew Gelman, John B Carlin, Hal S Stern, David B Dunson, Aki Vehtari, and Donald B Rubin. *Bayesian data analysis.* CRC press, 2013.

[15] Mike West. *Hyperparameter estimation in Dirichlet process mixture models.* Duke University ISDS Discussion Paper# 92-A03, 1992.

[16] Edward Meeds and Sam Roweis. Nonparametric bayesian biclustering. Technical report, Citeseer, 2007.